Science and Technology Facilities Council

Scientific Computing

Measuring the Eclipse with a micro:bit

Total solar eclipse over Madras, Oregon, 2017. This is a composite image where each orange shape is the sun at a different moment in time, showing the progression from partial eclipse through totality and back again.
Credit: NASA - science.nasa.gov/eclipses

Introduction

During a solar eclipse, the sky slowly gets darker because the moon moves infront of the sun! In this activity you will programme your micro:bit to measure how bright it is and save that data so you can look at it as a graph afterwards.

Micro:bits make this really easy! The tiny lights on your micro:bit (called LEDs) can also work the other way around and *detect* light, not just make it. So your micro:bit already has a light sensor built in - no extra parts needed!

As well as using coding to measure eclipses, we use the same types of programs to monitor the environment in our computer rooms and laboratories. It's important to keep all our scientists, computer scientists, engineers and technicians safe, as well as to understand the results of our experiments!

If you don't have a micro:bit, you can still do the coding for yourself, simulate what an eclipse might look like using the micro:bit simulator in MakeCode, and look at the data collected!

What You Will Need

- BBC micro:bit
- USB cable
- Battery pack
- A computer with a browser
- A torch

Part 1 - Build a Light Sensor Display

Step 1 - Open MakeCode

Open a browser and go to makecode.microbit.org. Click **New Project** and call it something like `eclipse sensor`.

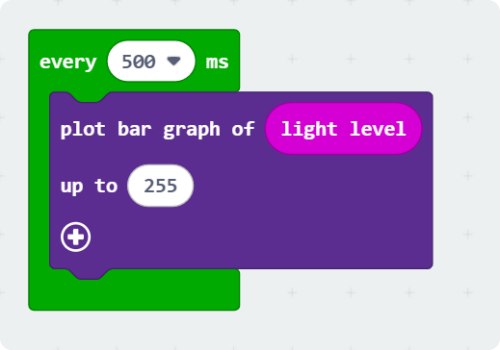
Step 2 - Add the Bar Graph Block

We are going to use a timed block that runs automatically every 500 milliseconds (that is twice in one second).

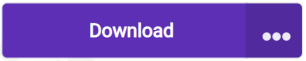
1. Open the **Loops** drawer and find the **every (500) ms** block. Drag it onto the screen.
2. Open the **LED** drawer and find **plot bar graph of (0) up to (0)**. Drag it inside the every block.
3. Go to the **Input** drawer and choose **light level**. Snap it to the first **0** on the bar graph block.
4. Change the second **0** (the "up to" part) to **255**.

> **Why 255?** The micro:bit measures light on a scale from **0** (completely dark) to **255** (as bright as it can detect). That range comes from how computers store numbers: the smallest unit of memory, called a *byte*, can hold exactly 256 different values - so counting from 0 gives you 0 to 255. Setting "up to 255" tells the bar graph what a full reading looks like, so the bars always show where the current light level sits on that scale.

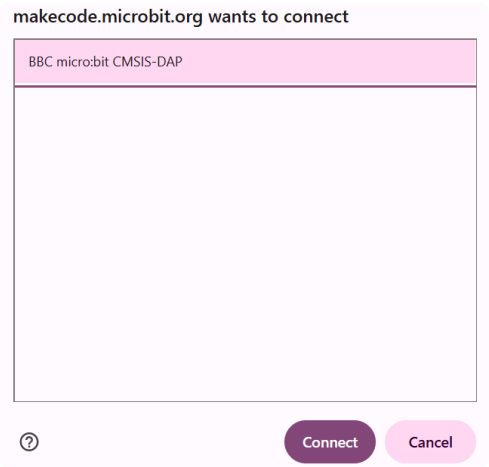
Your code should look like this:



Now send the code to your micro:bit by pairing it with the laptop.

1. Plug your micro:bit into the computer using the USB cable.
2. In MakeCode, click the 3 dots next to the 

button.
3. Choose **Connect device** from the menu that appears.
4. A box will pop up, click **Next** and then click **Pair**.
5. Another box will appear showing a list of devices. Click on **BBC micro:bit** and then click **Connect**.



6. MakeCode will now be paired with your micro:bit. Click **Download**.
7. The yellow light on the back of the micro:bit will flash while it is being programmed. Once it has stopped your code is now on the micro:bit!
8. You can now unplug your micro:bit from the laptop and plug in the battery pack.

> **Tip:** Once you have paired your micro:bit, you do not need to do steps 2–5 again. Next time, just click **Download** and your code will go straight onto your micro:bit.

Step 3 - Try It Out

Hold your micro:bit flat with the LED screen facing up.

- Shine a torch at the front of the micro:bit - the bars should go up.
- Cover the front with your hand - the bars should drop.
- Experiement with moving the torch closer and further away.

Can you make all the bars light up? Can you make them all disappear?

What about the eclipse?

> If you took your micro:bit outside during the eclipse, what would you expect to see happen to the bars?

The bar graph is great for watching the light change in real time - but the bars change as soon as the light changes again. To **measure** the eclipse properly, you need to **save the data** so you can look back at it as a **graph**.

That is what Part 2 is for.

Part 2 - Record Your Data

To save data on the micro:bit you need a special extension called **Data Logger**.

Step 4 - Add the Data Logger Extension

1. In MakeCode, click the **Extensions** drawer.
2. Search for `data logger` and click on it to add it. It is probably at the top already.
3. You should now see a new **Data Logger** drawer in the blocks menu.

Step 5 - Set Up the Column

Before you log any data, you need to tell the micro:bit what you are measuring. This goes in the **on start** block - it runs once when the micro:bit turns on.

1. Find the **on start** block (it may already be on your screen, if not have a look in **basic**).
2. Open the **Data Logger** drawer and find the **set columns** block. Drag it inside **on start**.
3. Click the text in the block and type `Light level`.

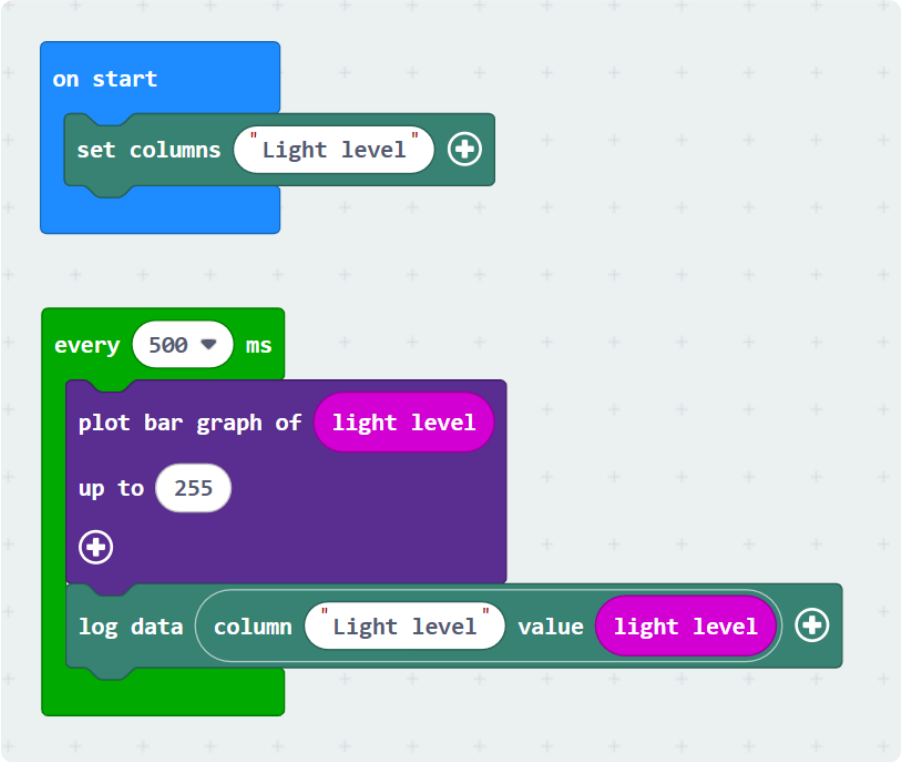
This block should now look like this:

Step 6 - Log the Data

Now you need to add the logging block inside your **every 500 ms** loop, right after the bar graph block.

1. From the **Data Logger** drawer, find the **log data column (value)** block. Drag it below your bar graph block.
2. Set the column name to `Light level` (it should match what you typed in Step 5).
3. Click the value gap, go to **Input**, and choose **light level**.

Your full code should now look like this:



Unplug the battery pack and plug your micro:bit back in to your laptop, then click **Download**. Once the yellow light on the back stops flashing the new code is on!

Step 7 - Try Recording Light Data

- Unplug the micro:bit and plug in the battery pack.
- Now experiement with the micro:bit again.
- Shine a torch at it, cover it with your hand, move the torch away.
- Do this for about a minute so you collect some interesting data.
- When you are done, plug the micro:bit back in.

Step 8 - Look at Your Data

When you plug the micro:bit back in, open the **MICROBIT** drive on the computer (it appears like a USB stick).

Inside you will find a file called **MY_DATA.HTM**. Double-click it to open it.

- You will see your light level readings listed as a table.
- Look for **Chart type** and next to it is a button that says **None**, click it and change it to **Line graph**.
- You should see a graph of how the light level changed over time!

> Can you see the moments when you covered the sensor or shone the torch at it?

Part 3 - The Eclipse

Now you are ready to use your micro:bit to measure the eclipse for real!

Step 9 - Head Outside

- Make sure your micro:bit is unplugged from the laptop and running on battery power.
- Place it front-side up somewhere flat, pointing at the open sky.
- Keep it away from shadows cast by you, a tree, or a building.
- Leave it running throughout the eclipse. The data logger will quietly save a reading every 500 ms without you needing to do anything!

Step 10 - After the Eclipse

- Bring your micro:bit back inside and plug it into your computer.
- Open the **MICROBIT** drive and find **MY_DATA.HTM** again.
- Open it and again switch to **Line graph** view.

You should see the light level drop during the eclipse and come back up afterwards - the shape of that dip is the eclipse, captured by your own sensor!

The Transit Method

That dip in your graph is not just a cool way to see the eclipse - it is the same technique astronomers use to discover planets around other stars!

When a planet passes in front of its star, it blocks a tiny fraction of the star's light, causing a small dip in brightness. By measuring how deep the dip is and how long it lasts, astronomers can work out the size of the planet and how long its year is. This is called the **transit method**.

Tips and Tricks

- **Start recording before the eclipse begins**. The more background data you collect beforehand, the clearer the dip will look on your graph.
- **The bars might not disappear completely**. Even losing half the bars during the eclipse is a real result - unless you are in the path of a total eclipse, the sky will get darker but not completely black.
- **Each time you download new code, the saved data is wiped**. Make sure your code is finalised before you go outside. Do not re-download to the micro:bit after you have started collecting data or you will lose it.
- **Want to do more?**. Have a look inside the **Data Logger** drawer. How could you use buttons to stop and restart the data logging?

Find Out More

Science and Technology Facilities Council (STFC)

STFC is one of nine UK Research and Innovation (UKRI) research councils, delivering world-leading research capabilities across astronomy, physics, space science, and computational science, with major campuses at Harwell and Daresbury.

STFC Scientific Computing

An international centre of excellence for advanced computing expertise and digital research infrastructure, providing the tools and platforms that power modern research. This includes the JASMIN Super-Data-Cluster and the LHC Tier-1 Data Centre. This activity was developed by the STFC Scientific Computing Public Engagement team.

STFC Public Engagement

STFC runs a programme of public engagement activities to inspire people with its research and technology, from school workshops and free teaching resources to STEM badge programmes with Girlguiding and Scouts.